

# Age of Information with a Packet Deadline

Clement Kam\*, Sastry Kompella\*, Gam D. Nguyen\*, Jeffrey E. Wieselthier<sup>†</sup> and Anthony Ephremides<sup>‡</sup>

\*Information Technology Division, Naval Research Laboratory, Washington, DC

<sup>†</sup>Wieselthier Research, Silver Spring, MD

<sup>‡</sup>Electrical and Computer Engineering Department, University of Maryland, College Park, MD

**Abstract**—In this work, we study the freshness of a continually updated piece of information as observed at a remote monitor by analyzing the age of information metric. The age of information has been studied for a variety of different queuing systems. In this work, we introduce a packet deadline as a control mechanism and study its impact on the average age of information for an M/M/1/2 queueing system. We analyze the system for a fixed deadline and derive a mathematical expression for the average age. We numerically evaluate the expression and show the relationship of the age performance to that of the M/M/1/1 and M/M/1/2 systems. We show that the system with a deadline constraint can outperform both the M/M/1/1 and M/M/1/2 without such a deadline.

## I. INTRODUCTION

We consider applications in which the goal is to continually communicate the most updated state of some time-varying process to a monitor. For example, a device regularly transmits packets containing some status (e.g., sensor data, list of neighboring nodes) to a network manager such that the observed status at the network manager stays relatively fresh at all times. Specifically, we look at a recently proposed metric known as the *status age* or the *age of information* for a system in which updates randomly pass through a queue. The *age* at the time of observation is defined as the current (observation) time minus the time at which the observed state was generated, and it directly describes this objective of achieving timely updating in a way that traditional metrics (e.g., delay, throughput) do not [1]–[3].

Research on the age metric has focused on optimizing the performance of systems that are modeled by different types of queues, with various arrival/departure processes, number of servers, and queue capacities. In particular, it was shown that deterministic arrival and departure processes achieve a lower average age than memoryless processes [1]. Also, the average age decreases as the number of servers increases [4], [5]. In addition, the age decreases as the queue capacity decreases or when packets in the queue are replaced with newer packets [6].

In addition to different queue models, we would like to uncover and understand other mechanisms for optimizing the age for different queues. In this work, we study the age metric when imposing a deadline on data packets that are waiting in a queue, such that they are dropped from the system when the deadline expires. Intuitively, a deadline that is too short would have more packets expiring, leading to less frequent updates at the monitor and a larger average age. However, a deadline that is too long would not discard packets that grow very stale

in the queue, resulting in the inefficient use of server resources on old packets, leading to an increase in the average age.

In this work, we derive the average age for an M/M/1/2 queue with a fixed deadline imposed on the packet in the queue. We choose an M/M/1/2 queue since the small queue achieves a relatively low average age. To reflect a common system constraint in which the packet in service cannot be dropped, we do not impose a deadline on the packet once its transmission has commenced. The case where a packet in service can be dropped may be of interest in certain systems but is outside of the scope of this work.

The analysis of the average age is challenging even for simple queues like the M/M/1/2. When we add the deadline requirement, the analysis is further complicated since we must account for packets that may exit the system without being served. Furthermore, we conduct our analysis for a fixed deadline, thus losing the memoryless property of the system that typically simplifies the model. However, we are able to get a mathematical expression for the average age, and we show through numerical evaluation that the imposition of deadline constraint can further optimize the system.

## II. SYSTEM MODEL

Similar to [6], we study a system in which a source transmits packets to a monitor through an M/M/1/2 queue, which has a total capacity of 1 packet in the queue and one packet in service. However, in this work, we consider that the packet waiting in the queue is subject to a deadline, such that if it waits in the queue for a time period longer than the deadline, it is dropped from the system and never enters service. If a packet enters the server before its deadline expires, it is guaranteed to be served and is never dropped. The case where packets in service can expire will be considered in future work. A plot of the age of information is shown in Figure 1, where transmissions occur at times  $t_1, t_2, \dots$ , and receptions at the monitor occur at times  $t'_1, t'_2, \dots$ .

We refer to the time between packet generations as the interarrival time  $X_i, i = 2, 3, \dots$ , which is equal to  $t_i - t_{i-1}$ . The interarrival times are modeled as random; consequently, the source does not have control over the exact times at which it can transmit updates. In our model, the  $X_i$ 's are i.i.d. exponential random variables with rate  $\lambda$ .

We call the time spent in the server by packet  $k$  the service time  $S_k, k = 1, 2, \dots$ , which is equal to  $t'_k - t_k$ . The service time  $S_k$  is modeled as exponential with rate  $\mu$ , and all the  $S_k$ 's

$$p_0 = \lim_{t \rightarrow \infty} \frac{\frac{\alpha(t)}{\alpha(t)+\beta(t)+\gamma(t)} \frac{1}{\alpha(t)} \sum_{j=1}^{\alpha(t)} V_{0j}}{\frac{\alpha(t)}{\alpha(t)+\beta(t)+\gamma(t)} \frac{1}{\alpha(t)} \sum_{j=1}^{\alpha(t)} V_{0j} + \frac{\beta(t)}{\alpha(t)+\beta(t)+\gamma(t)} \frac{1}{\beta(t)} \sum_{j=1}^{\beta(t)} V_{1j} + \frac{\gamma(t)}{\alpha(t)+\beta(t)+\gamma(t)} \frac{1}{\gamma(t)} \sum_{j=1}^{\gamma(t)} V_{2j}} \quad (2)$$

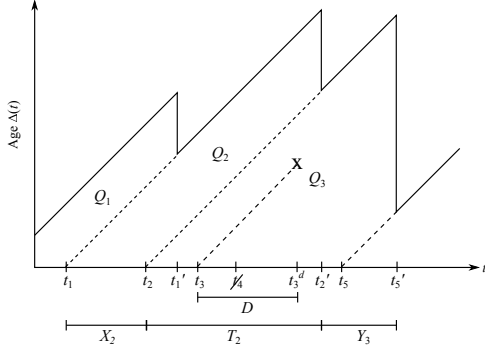


Fig. 1. Age of information for an M/M/1/2 system with deadlines.

are i.i.d. and independent of the  $X_i$ 's. The total time spent in the system from arrival to service is given by  $T_k$ ,  $k = 1, 2, \dots$ , where  $T_k = W_k + S_k$ , with  $W_k$  being the time spent waiting in the queue.

The *age of information* at time  $t$  is defined as  $\Delta(t) = t - u(t)$  [1], where  $u(t)$  is the timestamp of the most recent information at the receiver as of time  $t$ . Given this definition, we can see that the age increases linearly with  $t$  but is reset to a smaller value with each packet received that contains newer information, resulting in the sawtooth pattern shown in Figure 1.

We also define the *interdeparture time*  $Y_k$  as the time between the instants of complete service for the  $k-1$ st packet served and the  $k$ th packet served. This will be useful in the computation of the average age.

### III. AVERAGE AGE OF M/M/1/2 WITH A DEADLINE

To compute the average age for the M/M/1/2 system with a deadline, we apply the graphical argument as in [1]. We compute the area under the sawtooth curve using the sum of the trapezoids  $Q_k$ ,  $k \geq 0$  in Figure 1, where each trapezoid is associated with a unique packet that is successfully transmitted. Packets that are blocked by a full system (e.g., packet 4) and packets that are dropped due to an expiring deadline (e.g., packet 3) do not have trapezoids associated with them. Following the approach in [7], we compute the average area of trapezoid  $k$  by taking the area of the large isosceles triangle  $(1/2)(T_{k-1} + Y_k)^2$  and subtracting the area of the smaller isosceles triangle  $(1/2)T_k^2$ . The average age is given by  $\Delta = \lambda_e E[Q_k]$ , where  $\lambda_e$  is the effective arrival rate of packets that eventually complete service. Since  $T_{k-1}$  and  $T_k$  are equally distributed, the average age can be expressed as

$$\Delta_{M/M/1/2D} = \lambda_e \left( \frac{1}{2} E[Y_k^2] + E[T_{k-1} Y_k] \right). \quad (1)$$

We derive the terms  $\lambda_e$ ,  $E[Y_k^2]$ , and  $E[T_{k-1} Y_k]$  in the following sections.

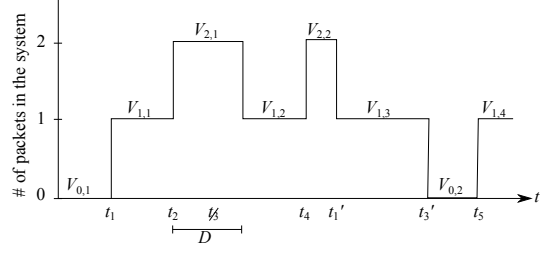


Fig. 2. Sample plot of time spent in each state.

#### A. Equilibrium Distribution of System State

Prior to deriving the terms in the average age expression, we first need to find the equilibrium distribution of the number of packets in the system, denoted as  $p_0$ ,  $p_1$ , and  $p_2$ . Because we are dealing with a fixed deadline, we lose the memorylessness of the system and cannot directly apply a Markovian analysis at any arbitrary time instance. Therefore, we apply a time averaging approach, making the assumption that time averages and ensemble averages are equal. Let  $V_{ij}$  be the time duration of the  $j$ th visit to the state in which there are  $i$  packets in the system. A sample plot of the time spent in each state is shown in Figure 2. We also define  $\alpha(t)$ ,  $\beta(t)$ , and  $\gamma(t)$  as the number of visits to state 0, 1, and 2, respectively. The percentage of time spent in state 0, for example, as  $t$  goes to infinity, is given by

$$p_0 = \lim_{t \rightarrow \infty} \frac{\sum_{j=1}^{\alpha(t)} V_{0j}}{t} = \lim_{t \rightarrow \infty} \frac{\sum_{j=1}^{\alpha(t)} V_{0j}}{\sum_{j=1}^{\alpha(t)} V_{0j} + \sum_{j=1}^{\beta(t)} V_{1j} + \sum_{j=1}^{\gamma(t)} V_{2j}}.$$

After some algebra, we get the expression in Equation 2 at the top of the page.

Assuming

$$\begin{aligned} \frac{\alpha(t)}{\alpha(t) + \beta(t) + \gamma(t)} &\rightarrow \pi_0, & \frac{\beta(t)}{\alpha(t) + \beta(t) + \gamma(t)} &\rightarrow \pi_1 \\ \frac{\gamma(t)}{\alpha(t) + \beta(t) + \gamma(t)} &\rightarrow \pi_2, & \frac{1}{\alpha(t)} \sum_{j=1}^{\alpha(t)} V_{0j} &\rightarrow E[V_0] \\ \frac{1}{\beta(t)} \sum_{j=1}^{\beta(t)} V_{1j} &\rightarrow E[V_1], & \frac{1}{\gamma(t)} \sum_{j=1}^{\gamma(t)} V_{2j} &\rightarrow E[V_2], \end{aligned}$$

where  $\pi_i$  is the percentage of visits that are made to the state  $i$  and  $E[V_i]$  is the average time spent in state  $i$ , we then have

$$p_0 = \frac{\pi_0 E[V_0]}{\pi_0 E[V_0] + \pi_1 E[V_1] + \pi_2 E[V_2]}$$

The probabilities  $p_1$  and  $p_2$  are derived in the same manner. To derive the  $\pi_i$ 's, we apply a Markov chain analysis, where

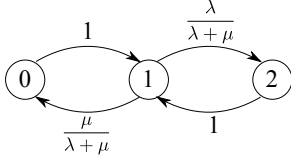


Fig. 3. Markov chain for the percentage of visits to state  $i$ .

the process of visits to the different states is represented by the Markov chain graph in Figure 3. Since the states 0 and 2 can only directly transition to 1, those transition probabilities are equal to 1. Since the interarrival times and service times are memoryless, the transition from 1 to 0 occurs when a service time is less than an interarrival time, which has probability  $\frac{\mu}{\lambda+\mu}$ . Likewise, the transition from 1 to 2 occurs when an interarrival time is less than a service time, which has probability  $\frac{\lambda}{\lambda+\mu}$ . Using the balance equations, we can show that the equilibrium distribution of visits is given by

$$\pi_0 = \frac{\mu}{2(\lambda + \mu)}, \pi_1 = \frac{1}{2}, \pi_2 = \frac{\lambda}{2(\lambda + \mu)}.$$

Lastly, we derive the average time spent in each state per visit. For  $V_0$ , the time spent is simply an interarrival time, so  $E[V_0] = 1/\lambda$ . For  $V_1$ , the time spent is given by

$$\begin{aligned} E[V_1] &= \Pr(X < S)E[X|X < S] + \Pr(S < X)E[S|S < X] \\ &= \frac{\lambda}{\lambda + \mu} \frac{1}{\lambda + \mu} + \frac{\mu}{\lambda + \mu} \frac{1}{\lambda + \mu} \\ &= \frac{1}{\lambda + \mu}. \end{aligned}$$

Finally,  $E[V_2]$  is the average service time given that the service time is less than  $D$  (if the residual service of the packet in the server is longer than the deadline  $D$ , the packet in the queue is dropped). This is given by

$$E[V_2] = \frac{1}{\mu}(1 - e^{-\mu D}).$$

We then have

$$\begin{aligned} p_0 &= \frac{\mu^2}{\mu^2 + \lambda\mu + \lambda^2(1 - e^{-\mu D})}, \\ p_1 &= \frac{\lambda\mu}{\mu^2 + \lambda\mu + \lambda^2(1 - e^{-\mu D})}, \\ p_2 &= \frac{\lambda^2(1 - e^{-\mu D})}{\mu^2 + \lambda\mu + \lambda^2(1 - e^{-\mu D})}. \end{aligned}$$

We note that the assumption of time averages equalling ensemble averages would be strengthened with a formal proof, but our simulations have indicated that this assumption holds.

### B. Effective Arrival Rate $\lambda_e$

To derive the effective arrival rate  $\lambda_e$ , we compute the probability that a packet is neither dropped due to deadline nor blocked by a full system. This is simply the probability that an arriving packet does not experience a residual service time greater than  $D$  nor does it see a full system:

$$\Pr(\text{not blocked or dropped}) = 1 - (p_1 e^{-\mu D} + p_2).$$

Thus the effective arrival rate is

$$\begin{aligned} \lambda_e &= \lambda(1 - (p_1 e^{-\mu D} + p_2)) \\ &= \lambda \left( \frac{\mu^2 + \lambda\mu(1 - e^{-\mu D})}{\mu^2 + \lambda\mu + \lambda^2(1 - e^{-\mu D})} \right) \end{aligned}$$

### C. Second Moment of the Interdeparture Time $E[Y_k^2]$

To compute the second moment of the interdeparture time, we condition on whether packet  $k-1$  departing the server leaves behind an empty system. We denote this event  $\psi$  and its complement  $\bar{\psi}$ . The event  $\psi$  occurs when a packet enters state 1 and the (residual) service time is less than an interarrival time. The event  $\bar{\psi}$  where packet  $k-1$  leaves behind a busy system occurs when the system enters state 2 and the (residual) service time of packet  $k-1$  is less than the deadline  $D$ . The probability of  $\psi$  is thus given by

$$\begin{aligned} \Pr(\psi) &= \Pr(\text{system last entered state 1} | \text{packet served}) \\ &= \frac{\pi_1 \frac{\mu}{\lambda+\mu}}{\pi_1 \frac{\mu}{\lambda+\mu} + \pi_2(1 - e^{-\mu D})} = \frac{\mu}{\mu + \lambda(1 - e^{-\mu D})} \end{aligned}$$

and  $\Pr(\bar{\psi}) = 1 - \Pr(\psi)$ .

The interdeparture time conditioned on  $\psi$  is the sum of a residual interarrival time and a service time. Taking the convolution of the two exponential random variables as in [7], we get

$$E[Y_k^2 | \psi] = \frac{2(\lambda^2 + \lambda\mu + \mu^2)}{\lambda^2 \mu^2}.$$

The interdeparture time conditioned on  $\bar{\psi}$  is simply a service time, so  $E[Y_k^2 | \bar{\psi}] = 2/\mu^2$ . Finally, to get  $E[Y_k^2]$  we substitute the conditional statistics in the following expression:

$$\begin{aligned} E[Y_k^2] &= E[Y_k^2 | \psi] \Pr(\psi) + E[Y_k^2 | \bar{\psi}] \Pr(\bar{\psi}) \\ &= \frac{2(\mu(\lambda^2 + \lambda\mu + \mu^2) + \lambda^3(1 - e^{-\mu D}))}{\lambda^2 \mu^2 (\mu + \lambda(1 - e^{-\mu D}))} \end{aligned}$$

### D. $E[T_{k-1}Y_k]$

Next we need to compute the quantity  $E[T_{k-1}Y_k]$ . Again, we condition on the events  $\psi$  and  $\bar{\psi}$ . In each case, the system time of packet  $k-1$  is conditionally independent of the interdeparture time for packet  $k$ , since the event  $\psi$  or  $\bar{\psi}$  determines whether  $Y_k$  is a residual interarrival time plus a service time or just a service time, independent of the just completed system time  $T_{k-1}$ . To compute  $E[T_{k-1}]$ , we first consider the waiting time for packet  $k-1$ . The waiting time of packet  $k-1$  is independent of the events  $\psi$  or  $\bar{\psi}$ . Thus we simply compute the expected waiting time for packets not blocked or dropped (denoted as “tx”):

$$\begin{aligned} E[W_{k-1} | \text{tx}] &= E[W_{k-1} | k-1 \text{ enters busy system, tx}] \\ &\quad \cdot \Pr(k-1 \text{ enters busy system} | \text{tx}) \\ &= E[S'_{k-n} | S'_{k-n} < D] \frac{\Pr(k-1 \text{ enters busy system, tx})}{\Pr(\text{tx})} \\ &= \frac{1}{1 - e^{-\mu D}} \left( \frac{1}{\mu} - \left( D + \frac{1}{\mu} \right) e^{-\mu D} \right) \frac{p_1(1 - e^{-\mu D})}{1 - (p_2 + p_1 e^{-\mu D})} \\ &= \left( \frac{1}{\mu} - \left( D + \frac{1}{\mu} \right) e^{-\mu D} \right) \frac{\lambda\mu}{\mu^2 + \lambda\mu(1 - e^{-\mu D})} \end{aligned}$$

Deriving the conditional expected service time given  $\psi$  is complicated by the deadline requirement. A seemingly straightforward derivation involves computing the service time distribution given that no packet is accepted in the time window of length  $D$  prior to the departure of the packet in service. However, this may occur when 1) no packets arrive in the time window, or 2) any packet that arrives within the time window is blocked by a packet already in the queue, which will itself expire just before the departure time. For case 2), we must also account for all events that ensure that there is a packet in the queue to do the blocking, by considering events where 1) no arrival occurs in the time window of length  $D$  prior, or 2) any arrival within the time window is itself blocked by some prior packet. We can see that this accounting procedure continues, making more complicated events to condition on.

Instead of attempting to account for all of these events, we look at the state of the system as a function of time, just as in Figure 2. At the start of a service, the system is in state 1 until either the packet is served and the system moves to state 0 or a packet arrives and the system moves to state 2. If the system moves to state 2, the packet in service cannot leave the system idle if it completes service in state 2. For a packet to finish service while in state 1, the system must first drop the waiting packet after a time  $D$  and move back to state 1, and the cycle starts over again. We compute the conditional expected service time given  $\psi$  and the number of packets  $l$  dropped during the service time:

$$\begin{aligned} E[S_{k-1}|\psi] &= \sum_{l=0}^{\infty} E[S_{k-1}|\psi, l \text{ dropped}] \Pr(l \text{ dropped}|\psi) \\ &= \sum_{l=0}^{\infty} \int_0^{\infty} s f(s|\psi, l \text{ dropped}) ds \Pr(l \text{ dropped}|\psi) \\ &= \sum_{l=0}^{\infty} \int_0^{\infty} s \frac{\Pr(\psi, l \text{ dropped}|s) f(s)}{\Pr(\psi, l \text{ dropped})} ds \\ &\quad \times \frac{\Pr(\psi, l \text{ dropped})}{\Pr(\psi)}. \end{aligned} \quad (3)$$

The probability  $\Pr(\psi, l \text{ dropped}|s)$  is that of the event where the end of a service time  $s$  falls after the sum of  $l$  interarrival times (for the  $l$  accepted arrivals while in state 1) and  $l$  periods of length  $D$  (for when the packets are dropped after in state 2), but before the arrival of the  $(l+1)$ st accepted packet. Let  $Z_l$  be the sum of the  $l$  interarrival times. We then have

$$\begin{aligned} \Pr(\psi, l \text{ dropped}|s) &= \Pr(Z_l + lD < s < Z_l + X_{l+1} + lD) \\ &= \Pr(s - lD - Z_l < X_{l+1}) \\ &= \int_0^{s-lD} e^{-\lambda(s-lD-z)} f_{Z_l}(z) dz \end{aligned}$$

for  $s > lD$ , zero otherwise. Using the PDF for the Erlang

distributed  $Z_l$  in Equation 3, we get

$$\begin{aligned} E[S_{k-1}|\psi] &= \frac{1}{\Pr(\psi)} \int_0^{\infty} s \left( e^{-\lambda s} + \sum_{l=1}^{\infty} e^{-\lambda(s-lD)} \right. \\ &\quad \times \left. \int_0^{s-lD} \frac{\lambda^l z^{l-1}}{(l-1)!} dz \right) \mu e^{-\mu s} ds \\ &= \frac{1}{\Pr(\psi)} \int_0^{\infty} s \left( e^{-\lambda s} + \sum_{l=1}^{\lfloor \frac{s}{D} \rfloor} \right. \\ &\quad \times \left. \frac{(\lambda(s-lD))^l e^{-\lambda(s-lD)}}{l!} \mathbb{1}\{s > lD\} \right) \\ &\quad \times \mu e^{-\mu s} ds \\ &\vdots \\ &= \frac{1}{\Pr(\psi)} \left( \frac{\mu}{(\lambda + \mu)^2} + \sum_{n=1}^{\infty} \sum_{l=1}^n \frac{\lambda^l \mu}{l!} \right. \\ &\quad \times \left. \int_{nD}^{(n+1)D} s(s-lD)^l e^{-(\lambda+\mu)s+\lambda lD} ds \right) \end{aligned}$$

For the conditional expected service time given  $\bar{\psi}$ , we can use the fact that the average service time is  $1/\mu$  to get  $E[S_{k-1}|\bar{\psi}] = (1/\mu - E[S_{k-1}|\psi] \Pr(\psi))/\Pr(\bar{\psi})$ .

As mentioned before, the system time for packet  $k-1$  and the interdeparture time for packet  $k$  are conditionally independent given  $\psi$  and  $\bar{\psi}$ . We now have all of the elements needed to evaluate the following expression:

$$\begin{aligned} E[T_{k-1}Y_k] &= E[T_{k-1}Y_k|\psi] \Pr(\psi) + E[T_{k-1}Y_k|\bar{\psi}] \Pr(\bar{\psi}) \\ &= \left( \frac{1}{\lambda} + \frac{1}{\mu} \right) E[T_{k-1}|\psi] \Pr(\psi) \\ &\quad + \frac{1}{\mu} E[T_{k-1}|\bar{\psi}] \Pr(\bar{\psi}) \\ &= \left( \frac{1}{\lambda} + \frac{1}{\mu} \right) (E[W_{k-1}|\text{tx}] + E[S_{k-1}|\psi]) \Pr(\psi) \\ &\quad + \frac{1}{\mu} (E[W_{k-1}|\text{tx}] + E[S_{k-1}|\bar{\psi}]) \Pr(\bar{\psi}) \end{aligned}$$

#### E. Average Age as $\lambda \rightarrow \infty$

We have now derived the terms necessary to compute the average age. If we take the limit of the average age expression as  $\lambda$  goes to infinity, we arrive at the following expression (omitting the derivation for brevity):

$$\lim_{\lambda \rightarrow \infty} \Delta_{M/M/1/2D} = \frac{3}{\mu} - \left( D + \frac{1}{\mu} \right) e^{-\mu D}.$$

Letting  $D \rightarrow \infty$  results in  $\Delta_{M/M/1/2D} = 3/\mu$ , which is the average age for the M/M/1/2 without deadline as  $\lambda$  goes to infinity, and setting  $D = 0$  results in  $\Delta_{M/M/1/2D} = 2/\mu$ , which is the average age for the M/M/1/1 without deadline as  $\lambda$  goes to infinity [7]. If we take the derivative with respect to  $D$ , we have

$$\frac{\partial}{\partial D} \lim_{\lambda \rightarrow \infty} \Delta_{M/M/1/2D} = -e^{-\mu D} + \mu \left( D - \frac{1}{\mu} \right)$$

which is always greater than 0, since  $e^{-\mu D}$  is at most 1 for  $D \geq 0$ . Therefore, for the case when  $\lambda$  approaches infinity, the minimum average age occurs for  $D = 0$ .

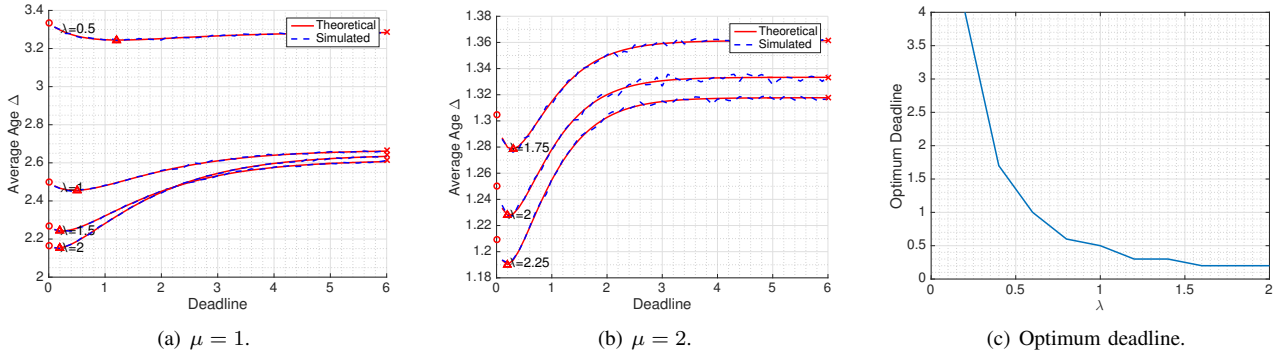


Fig. 4. (a) Average age vs. deadline for  $\mu = 1$ ,  $\lambda = 0.5, 1, 1.5$ , and  $2$ , (b) Average age vs. deadline for  $\mu = 2$ ,  $\lambda = 1.75, 2$ , and  $2.25$ . Minima are indicated by " $\triangle$ ", M/M/1/1 age indicated by " $\circ$ ", M/M/1/2 age indicated by " $\times$ ". (c) Optimum deadline vs.  $\lambda$  for  $\mu = 1$ .

#### IV. NUMERICAL RESULTS

We numerically evaluate the average age  $\Delta_{M/M/1/2D}$  for various service rates  $\mu$  and arrival rates  $\lambda$ , and plot the age vs. deadline in Figures 4(a) and 4(b) for  $\mu = 1$  and  $2$ , respectively. We also plot the age calculated from simulations of 10,000 samples averaged over 100 runs. The simulation agrees closely with theory, supporting the assumption in Section III-A that time averages and ensemble averages are equal. The minimum for each case of  $\lambda$  is marked with a " $\triangle$ ". In Figure 4(a), we observe that for small values of  $\lambda$ , after initially decreasing, the average age increases toward an asymptotic value as the deadline increases. Since packets arrive infrequently, there is less of a need to drop the packet in queue since it will not be replaced frequently. For larger values of  $\lambda$ , the average age starts to decrease with an increasing deadline, but then the age quickly starts to increase and then asymptotically approaches a limit. As  $\lambda$  continues to increase, the age decreases only slightly before increasing more as deadline increases, since a packet dropped from the queue is likely to be replaced immediately, keeping packets fresh. These results show that the deadline has a larger impact for higher  $\lambda$ . For Figure 4(b) where  $\mu = 2$ , the average age is lower than for  $\mu = 1$ , and the deadline has a more pronounced effect in lowering the age. This suggests that a deadline will have a more noticeable impact for systems that operate at a higher rate.

Note that when the deadline is set to 0, the system is equivalent to an M/M/1/1 system since no packet can wait in the queue. For a deadline equal to infinity, the system is equivalent to an M/M/1/2 system. We have also plotted the values of the age for these systems in Figures 4(a)-4(b) (" $\circ$ " for M/M/1/1, " $\times$ " for M/M/1/2). It is shown in [7] that the M/M/1/1 has a higher average age for small values of  $\rho = \lambda/\mu$ , but a lower average age for large values of  $\rho = \lambda/\mu$ . The intuition is that for lower arrival rates, the M/M/1/1 will have to wait longer after a packet departure for another packet to send, thus increasing the age. On the other hand, for higher arrival rates, the M/M/1/1 will typically be filled with a fresh packet shortly after a departure whereas the M/M/1/2 will have a packet in waiting that is typically older. Based on these results, we observe that the imposition of a packet deadline constraint is one way to transition between the M/M/1/1 and

M/M/1/2, at worst getting the best of both approaches, but in actuality improving upon them in most cases.

Figure 4(c), shows that the optimum value of deadline decreases as  $\lambda$  increases. This result is intuitively satisfying because a higher arrival rate implies that packets are dropped and replaced more frequently.

#### V. CONCLUSION AND FUTURE WORK

We have observed that the use of a packet deadline can add a new dimension to optimizing the age of information, thus improving the performance of real-time monitoring applications. We have provided a mathematical analysis of the average age expressed in terms of  $\lambda$ ,  $\mu$ , and  $D$ . Our numerical evaluation shows that the age approaches the M/M/1/1 and M/M/1/2 ages as the deadline approaches 0 and  $\infty$ , respectively, but there is also an optimum deadline that yields an even lower age. We also observe that the optimum deadline decreases for increasing arrival rate. Future directions for this work include considering larger queue capacities, adaptive deadlines, and packet management. We are particularly interested to see if a deadline will have any added value when the queue is capable of replacing packets with newer arriving packets.

#### ACKNOWLEDGMENT

This work was supported by the Office of Naval Research.

#### REFERENCES

- [1] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, Orlando, FL, Mar. 2012, pp. 2731–2735.
- [2] —, "Status updates through queues," in *Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, Mar. 2012, pp. 1–6.
- [3] R. D. Yates and S. Kaul, "Real-time status updating: Multiple sources," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Cambridge, MA, Jul. 2012, pp. 2666–2670.
- [4] C. Kam, S. Kompella, and A. Ephremides, "Age of information under random updates," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Istanbul, Turkey, Jul. 2013, pp. 66–70.
- [5] —, "Effect of message transmission diversity on status age," in *Information Theory (ISIT), 2014 IEEE International Symposium on*, June 2014, pp. 2411–2415.
- [6] M. Costa, M. Codreanu, and A. Ephremides, "Age of information with packet management," in *IEEE International Symposium on Information Theory (ISIT)*, June 2014, pp. 1583–1587.
- [7] —, "On the age of information in status update systems with packet management," *CoRR*, vol. abs/1506.08637, 2015. [Online]. Available: <http://arxiv.org/abs/1506.08637>